

value can, in turn, be passed on to other intelligence objects that continue to build on, or add to, the measurements so that performance across multiple computer, tiers, operating systems, applications, etc., is achieved.

[55] Ultimately, the utilization value, or values, is passed on to other processes which can display the result of the combined measurements to a human user, use the result to derive other results, use the result to automate optimization of the n-tiered system, or use the result for other purposes. One aspect of the invention provides for redirecting processes and interconnections on the network based on the assessed utilization values of the computers, or nodes, in order to improve, or optimize, network performance. The processes that perform the redirection are referred to as “process redirection objects” (PROSE).

[56] Note that although the invention is sometimes discussed with respect to a multi-tiered server arrangement that any arrangement of servers, computers, digital processors, etc., is possible. The term “processing device” is used to refer to any hardware capable of performing a function on data. Processing devices include servers, computers, digital processors, storage devices, network devices, input/output devices, etc. Networks need not be in a multi-tiered arrangement of processing devices but can use any arrangement, topology, interconnection, etc. Any type of physical or logical organization of a network is adaptable for use with the present invention.

[57] Fig. 2B illustrates one possible arrangement of more specific components of the present invention. Note that the term “component” as used in this specification includes any type of processing device, hardware or software that may exist or may be executed within or by a digital processor or system.

[58] Systems such as those illustrated in Figs. 1, 2A and 2B, along with virtually any type of networked system, can be provided with IOs. In a preferred embodiment, the IOs are installed on each server in the network in a distributed peer-to-peer architecture. The IOs, along with aggregation software, discussed below, measure real-time behavior of the servers components, resources, etc. to achieve an overall measure of the behavior and performance of the network. A preferred embodiment rates and aggregates network components using a system-wide model discussed in the related applications discussed, above.

[59] The preferred embodiment collects data on low-level system and network parameters such as CPU utilization, network utilization, latency, etc. The data is produced and shared in small four-byte values. In a hierarchy set up by an administrator, or automatically configured by the system, a value is combined with other values to achieve a composite value. The composite value is then passed along the hierarchy and used to obtain

further composited values so that overall system performance is ultimately provided in the composited values.

[60] One problem with using composite values that are maintained from node-to-node is that a network may have multiple different hardware and software components at different points in the system. Typically, hardware for executing software, combined with operating system software is referred to as a “platform.” For purposes of this application, “platform” refers to any combination of hardware and software, or portion thereof, used to allow other software, or processes, including the nodes of the present invention, to execute. In this sense, any given platform, may change frequently over time as processes are terminated and started, hardware is reconfigured, etc.

[61] Fig. 2C illustrates a network system with multiple platforms.

[62] In Fig. 2C, network system 150 includes various components, including server computers shown as blocks. Each server computer can use different hardware such as different numbers and types of central processing units (CPUs), amounts and types of memory, architecture, peripherals, etc.

[63] Different software can also be used. For example, server 152 executes the Windows 2000 operating system while server 154 executes Windows NT and server 156 executes Linux. Other servers are shown executing different application programs and operating systems. Naturally, any number and type of hardware and software can be employed. Further, the network configuration can vary widely from that shown in Fig. 2C. In general, any network configuration can be used with the present invention.

[64] Values which are intended to convey the same meaning may, in fact, have different meanings in association with different platforms. For example, processor speed, instructions per second, interrupts, input/output operations, number and priority of threads, number and type of forked processes, memory management, block allocation, etc. have different effective meanings depending on the platform that is being measured or reported. Thus, it is important to adjust, resolve, normalize or homogenize, values with respect to the different platforms so that the values can be combined, or composited, as described, below, for more effective reporting and monitoring.

[65] For example, one parameter that is accumulated is the number of blocks allocated by the operating system over time. This parameter is meaningful since it reflects the memory utilization of a component, or platform, in the system. However, different operating systems may use different size blocks so just keeping track of the number of blocks would give inaccurate results. Other factors which lead to incompatible comparisons and use

of the block allocation parameter is the number of blocks available in the system, the overhead involved (e.g., processor cycles, memory, etc.) in performing the block allocation, etc.

[66] One embodiment of the invention stores, e.g., the memory block size for different platforms. When a block allocation parameter is received from a platform (e.g., as part of a LNV or CNV, discussed, below) the parameter is adjusted according to the block size. For example, where a block allocation parameter comes from a platform where the block size is one-half the block size on a platform executing a node that receives the parameter, the node adjusts the parameter by a factor of two to account for the difference between the two platforms. In this manner, the parameter values can be combined, or composited, to achieve the benefits discussed below.

[67] A network set up with the IOs and other monitoring, analysis and optimization tools as discussed herein is referred to as a Dynamic Aggregate System Process Optimization (DASPO) network. There are three basic phases of operating a DASPO to achieve network improvement or optimization. These phases are (1) set-up, (2) analysis and (3) optimization. In a preferred embodiment, the system of the present invention provides various user tools, including console interfaces, to allow a human user to participate in the different phases. However, provision is also made for automating the different phases to varying degrees.

[68] The operation and implementation of the three phases is heavily dependent on the system-wide model employed by the present invention. The system-wide model is discussed, below, in connection with the three phases and user interfaces for controlling the three phases.

Set-Up

[69] There are five basic steps in setting up a DASPO network, as follows:
 Define Node Resource Pools (NRPs)
 Add Nodes
 Install Intelligence Objects on Selected Nodes
 Define Functional Resource Pools (FRPs); and
 Establish Connectivity and Data Flow

[70] Fig. 3A illustrates a user interface display to set up node resource pools. In Fig. 3A, node pools are displayed as ovals with labels. NRPs are used to group nodes for organizational purposes. NRPs are used in place of the tier illustration approach of Figs. 1A